

## PRACTICAL 1

AIM : 1.1 : Create “Custom Message” application. That will display “Custom Message” in the middle of the screen.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Custom Message"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity file:

```
package com.example.new_practs_228
```

```
import android.os.Bundle
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
```

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
```

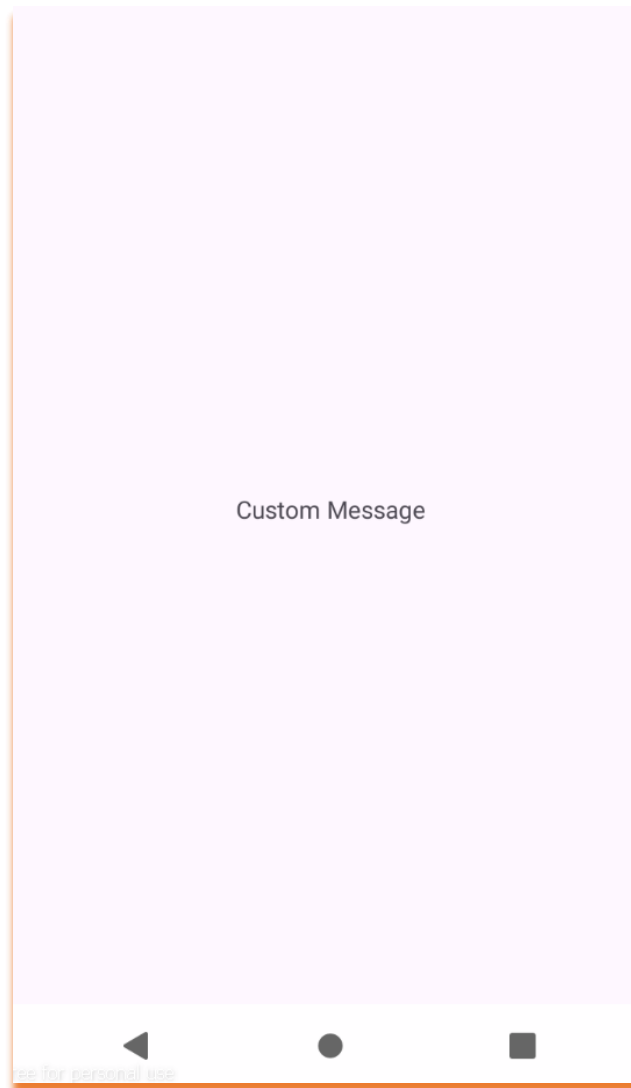
```

ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
    val systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars())
    v.setPadding(systemBars.left, systemBars.top, systemBars.right,
systemBars.bottom)
    insets
}
}
}

```

OUTPUT:

\



1.2: Design an application representing a simple calculator.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editTextText2"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:inputType="numberDecimal"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintWidth_default="spread"
        android:layout_marginTop="16dp"/>

    <EditText
        android:id="@+id/editTextText3"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="Enter second number"
        android:inputType="numberDecimal"
        app:layout_constraintTop_toBottomOf="@+id/editTextText2"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="16dp"/>

    <TextView
        android:id="@+id/textView3"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="Result: "
        android:textSize="20sp"
        app:layout_constraintTop_toBottomOf="@+id/editTextText3"
```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="16dp"/>
```

<Button

```
android:id="@+id/button"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:text="Add"
app:layout_constraintTop_toBottomOf="@+id/textView3"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="16dp"/>
```

<Button

```
android:id="@+id/button2"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:text="Subtract"
app:layout_constraintTop_toBottomOf="@+id/button"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="16dp"/>
```

<Button

```
android:id="@+id/button3"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:text="Multiply"
app:layout_constraintTop_toBottomOf="@+id/button2"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="16dp"/>
```

<Button

```
android:id="@+id/button4"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:text="Divide"
app:layout_constraintTop_toBottomOf="@+id/button3"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="16dp"/>
```

```

<Button
    android:id="@+id/button5"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="AC"
    app:layout_constraintTop_toBottomOf="@+id/button4"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="10dp"/>

<TextView
    android:id="@+id/textView"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Log Display: "
    android:textSize="20sp"
    app:layout_constraintTop_toBottomOf="@+id/button5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity file:

```
package com.example.basic_calc_kotlin
```

```

import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import android.widget.EditText
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

```

```

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        val etFirstNumber = findViewById<EditText>(R.id.editTextText2)
        val etSecondNumber =
            findViewById<EditText>(R.id.editTextText3)
    }
}

```

```
val tvResult = findViewById<TextView>(R.id.textView3)
```

```
val btnAdd = findViewById<Button>(R.id.button)  
val btnSubtract = findViewById<Button>(R.id.button2)  
val btnMultiply = findViewById<Button>(R.id.button3)  
val btnDivide = findViewById<Button>(R.id.button4)  
val btnac = findViewById<Button>(R.id.button5)
```

```
// Add
```

```
btnAdd.setOnClickListener {  
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()  
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()  
    if (num1 != null && num2 != null) {  
        val result = num1 + num2  
        tvResult.text = "Result: $result"  
    } else {  
        tvResult.text = "Please enter valid numbers"  
    }  
}
```

```
// Subtract
```

```
btnSubtract.setOnClickListener {  
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()  
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()  
    if (num1 != null && num2 != null) {  
        val result = num1 - num2  
        tvResult.text = "Result: $result"  
    } else {  
        tvResult.text = "Please enter valid numbers"  
    }  
}
```

```
// Multiply
```

```
btnMultiply.setOnClickListener {  
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()  
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()  
    if (num1 != null && num2 != null) {  
        val result = num1 * num2  
        tvResult.text = "Result: $result"  
    } else {  
        tvResult.text = "Please enter valid numbers"  
    }  
}
```

```

// Divide
btnDivide.setOnClickListener {
    val num1 = etFirstNumber.text.toString().toDoubleOrNull()
    val num2 = etSecondNumber.text.toString().toDoubleOrNull()
    if (num1 != null && num2 != null) {
        if (num2 != 0.0) {
            val result = num1 / num2
            tvResult.text = "Result: $result"
        } else {
            tvResult.text = "Cannot divide by zero"
        }
    } else {
        tvResult.text = "Please enter valid numbers"
    }
}
//ac
btnac.setOnClickListener {

}
//activity life cycle

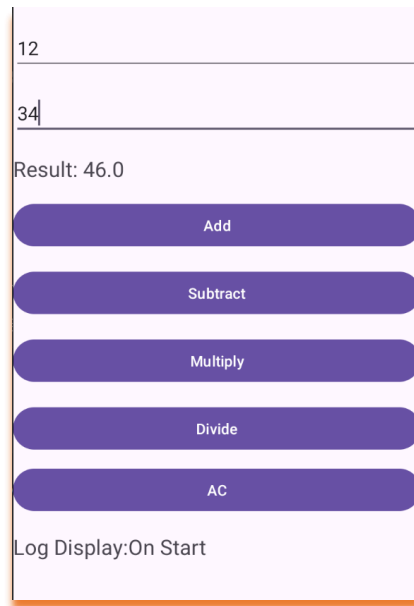
```

```

ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
    val systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars())
    v.setPadding(systemBars.left, systemBars.top, systemBars.right,
systemBars.bottom)
    insets
}
}
}

```

OUTPUT:



1.3: Create a login application with following features:

1. Successful Login message in TextView with Green background if Username & password is correct
2. Failure message in TextView with Red background if Username or password is incorrect.
3. Disable login Button after three wrong login attempts.
4. Close application if user selects Cancel Button.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/main"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">
<!-- User name -->
<EditText
    android:id="@+id/editTextText"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:hint="Enter Username"
    app:layout_constraintTop_toTopOf="parent"
```



```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:inputType="text"
        android:layout_marginTop="16dp" />
<!-- password -->
<EditText
    android:id="@+id/editTextText2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:hint="Enter Password"
    android:inputType="textPassword"
    app:layout_constraintTop_toBottomOf="@+id/editTextText"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
<!-- login button -->
<Button
    android:id="@+id/button"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Login"
    app:layout_constraintTop_toBottomOf="@+id/editTextText2"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
<!-- Cancel button -->
<Button
    android:id="@+id/button2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Cancel"
    app:layout_constraintTop_toBottomOf="@+id/button"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp" />
<!-- Result -->
<TextView
    android:id="@+id/textView"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text=""
    android:textSize="20sp"
    android:gravity="center"

```

```

        android:visibility="gone"
        app:layout_constraintTop_toBottomOf="@+id/button2"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="16dp" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity file:

```
package com.example.p1_3_login
```

```

import android.graphics.Color
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
import kotlin.system.exitProcess

```

```

class MainActivity : AppCompatActivity() {
    private var attempts = 0
    private val maxAttempts = 3
    private val correctUsername = "admin"
    private val correctPassword = "password123"
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        val etUsername = findViewById<EditText>(R.id.editTextText)
        val etPassword = findViewById<EditText>(R.id.editTextText2)
        val btnLogin = findViewById<Button>(R.id.button)
        val btnCancel = findViewById<Button>(R.id.button2)
        val tvResult = findViewById<TextView>(R.id.textView)
        btnLogin.setOnClickListener {
            val username = etUsername.text.toString()
            val password = etPassword.text.toString()
            // Check if login is successful
            if (username == correctUsername && password ==
correctPassword) {
                tvResult.text = "Login Successful!"
                tvResult.setBackgroundColor(Color.GREEN)
            }
        }
    }
}

```

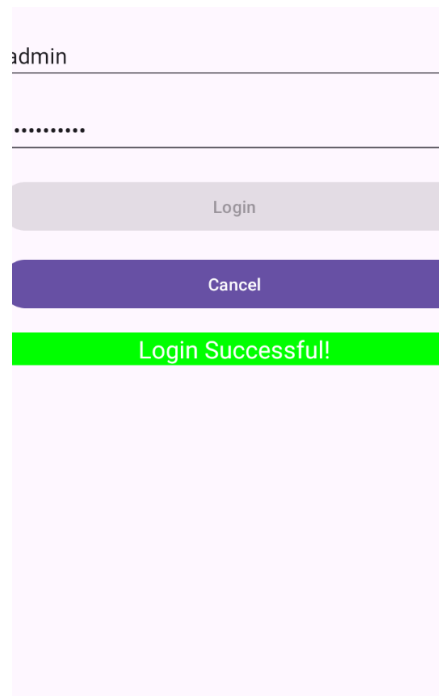
```

        tvResult.setTextColor(Color.WHITE)
        tvResult.visibility = TextView.VISIBLE
        btnLogin.isEnabled = false
    } else {
        attempts++
        if (attempts >= maxAttempts) {
            btnLogin.isEnabled = false
            tvResult.text = "Login Disabled! Too many attempts."
            tvResult.setBackgroundColor(Color.RED)
            tvResult.setTextColor(Color.WHITE)
        } else {
            tvResult.text = "Login Failed! Attempts remaining:
${maxAttempts - attempts}"
            tvResult.setBackgroundColor(Color.RED)
            tvResult.setTextColor(Color.WHITE)
        }
        tvResult.visibility = TextView.VISIBLE
    }
}
btnCancel.setOnClickListener {
    // Close the application when Cancel button is pressed
    finishAffinity()
    exitProcess(0)
}

ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.mai
n)) { v, insets ->
    val systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars())
    v.setPadding(systemBars.left, systemBars.top, systemBars.right,
systemBars.bottom)
    insets
}
}
}

```

OUTPUT:



1.4: Create android application to demonstrate life cycle of activity.

CODE:

XML FILE:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="Lifecycle Logs"
        android:textSize="18sp"
        android:gravity="center"
```

```
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="16dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity file:

```
package com.example.activity_lifecycle
```

```
import android.os.Bundle
import android.util.Log
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
```

```
class MainActivity : AppCompatActivity() {
    private lateinit var tvLifecycle: TextView
    // Tag for logging
    private val TAG = "ActivityLifecycle"
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        tvLifecycle = findViewById(R.id.textView)

        // Log and display onCreate state
        logAndDisplay("onCreate")
    }

    override fun onStart() {
        super.onStart()
        // Log and display onStart state
        logAndDisplay("onStart")
    }
    override fun onResume() {
        super.onResume()
        // Log and display onResume state
        logAndDisplay("onResume")
    }

    override fun onPause() {
```

```

        super.onPause()
        // Log and display onPause state
        logAndDisplay("onPause")
    }

    override fun onStop() {
        super.onStop()
        // Log and display onStop state
        logAndDisplay("onStop")
    }

    override fun onRestart() {
        super.onRestart()
        // Log and display onRestart state
        logAndDisplay("onRestart")
    }

    override fun onDestroy() {
        super.onDestroy()
        // Log and display onDestroy state
        logAndDisplay("onDestroy")
    }
    // Method to log and update the TextView with the current lifecycle
state
    private fun logAndDisplay(state: String) {
        Log.d(TAG, state)
        tvLifecycle.text = "Current state: $state"
    }

}

```

OUTPUT:

2024-10-07 10:45:57.632	2958-2994	ProfileInstaller	com.example.activity_lifecycle	D	Installing profile for com.example.acti
2024-10-07 10:46:01.222	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onPause
2024-10-07 10:46:01.710	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onStop
2024-10-07 10:46:04.544	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onRestart
2024-10-07 10:46:04.545	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onStart
2024-10-07 10:46:04.546	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onResume
2024-10-07 10:46:04.586	2958-2980	OpenGLRenderer	com.example.activity_lifecycle	E	Unable to match the desired swap behavi
2024-10-07 10:46:11.996	2958-2971	System	com.example.activity_lifecycle	W	A resource failed to call close.
2024-10-07 10:46:52.593	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onPause
2024-10-07 10:46:53.206	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onStop
2024-10-07 10:46:59.427	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onRestart
2024-10-07 10:46:59.430	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onStart
2024-10-07 10:46:59.436	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onResume
2024-10-07 10:46:59.473	2958-2980	OpenGLRenderer	com.example.activity_lifecycle	E	Unable to match the desired swap behavi
2024-10-07 10:47:06.982	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onPause
2024-10-07 10:47:07.450	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onStop
2024-10-07 10:47:11.052	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onRestart
2024-10-07 10:47:11.054	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onStart
2024-10-07 10:47:11.056	2958-2958	ActivityLifecycle	com.example.activity_lifecycle	D	onResume
2024-10-07 10:47:11.101	2958-2980	OpenGLRenderer	com.example.activity_lifecycle	E	Unable to match the desired swap behavi